In Quest of Performance Metrics for Intelligent Systems—A Challenge that Cannot be Met with Existing Methods

Lotfi A. Zadeh[*]

As we move further into the realm of intelligent systems, the problem of devising performance metrics for assessing machine intelligence looms larger and larger in importance. The problem is there, but does it have a solution? A somewhat unorthodox view which is articulated in the following is that (a) complete solution is beyond the reach of existing methods; and (b) that a prerequisite to solving the problem is a better understanding of a broader problem, namely, the basic problem of concept definability. To this end, what is presented in the following is a sketch of what may be called a theory of hierarchical definability, or THD for short.

In science, and especially in natural sciences and mathematics, there is a long-standing tradition of expecting that concepts be defined clearly and precisely. But as we move from the natural sciences to the sciences of the artificial, two basic problems came into view.

The first problem relates to the need to formulate our definitions in ways that can be understood by a machine. For example, if I command a household robot to take the dishes off the table, I must define what I mean by "take the dishes off the table." Or, if I instruct a machine to summarize a document, I must define what I mean by a summary. And, how can I assess the MIQ (Machine IQ) of a machine that executes my commands?

The second problem is that we encounter, much more frequently than in the past, concepts which do not lend themselves to precise definition. Among familiar examples of such concepts are intelligence, creativity, autonomy, adaptivity, relevance, robustness and causality.

We have been largely unsuccessful in formulating operational definitions of concepts of this nature. Why?

A view that is advanced in the following is that the primary reason for the lack of success is that the concepts in question, and many like them, are intrinsically fuzzy, that is, are a matter of degree. Thus, when we try to define such concepts within the conceptual framework of classical, bivalent logic, we encounter a fundamental incompatibility—an incompatibility between crispness of definitions and fuzziness of the concepts we try to define.

Report Documentation Page

1. REPORT DATE **AUG 2002**	2. REPORT TYPE	3. DATES COVERED **00-00-2002 to 00-00-2002**

4. TITLE AND SUBTITLE **In Quest of Performance Metrics for Intelligent Systems-A Challenge that Cannot be Met with Existing Methods**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of California,Department of EECS,Computer Science Division and the Electronics Research Laboratory,Berkeley,CA,94720-1776**	8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT **Approved for public release; distribution unlimited**

13. SUPPLEMENTARY NOTES **Proceedings of the 2002 Performance Metrics for Intelligent Systems Workshop (PerMIS ?02), Gaithersburg, MD on August 13-15, 2002**

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**4**	

Viewed in a slightly different perspective, the problem relates to the inadequate expressive power of the definition languages which are at our disposal, namely, the natural language and the language of bivalent logic. What this implies is that, to solve the problem, we have to add languages with higher expressive power to our repertoire of definition languages. This is the basic idea that underlies the theory of hierarchical definability, THD.

In THD, the languages that we add are based on fuzzy logic since they must be capable of serving as definition languages for fuzzy concepts. More specifically, the definition languages in THD form a hierarchy represented as (NL, C, F, F.G, PNL), where NL is the lowest member in terms of expressive power, and PNL (Precisiated Natural Language) is the highest. It is understood that every member of the hierarchy subsumes those below it.

The C definition language is the language of mathematical analysis, probability theory and bivalent logic. This is the language that we learn when we take courses in mathematics, probability theory and logic. The F language is the language of fuzzy logic without granulation, and the F.G language is the language of fuzzy logic with granulation. PNL (Precisiated Natural Language) is fuzzy-logic-based language with maximal expressive power.

A simple analogy may be of help. In my progression of learning, I start with my knowledge of a natural language. After entering a university and taking courses in mathematics, I add to NL my knowledge of C. At this stage, I can use the union of NL and C as a definition language. Then, I take a course in fuzzy logic. In this course, first I learn F, then F.G and finally PNL. At the end, I can use PNL as a definition language, with the understanding that PNL subsumes all languages below it in the hierarchy.

What is PNL? The basic idea in PNL is that a proposition, p, in a natural language, NL, may be precisiated through translation into a precisiation language. In the case of PNL, the precisiation language is the Generalized Constraint Language (GCL). A generic generalized constraint is represented as Z isr R, where Z is the constrained variable, R is the constraining relation and r is a discrete-valued indexing variable whose values define the ways in which R constrains Z. The principal types of constraints are: possibilistic (r=blank); veristic (r=v); probabilistic (r=p); random set (r=rs); usuality (r=u); fuzzy graph (r=fg); and Pawlak set (r=ps). The rationale for constructing a large variety of constraints is that conventional crisp constraints are incapable of representing the meaning of propositions expressed in a natural language—most of which are intrinsically imprecise—in a form that lends itself to computation.

The elements of GCL are composite generalized constraints that are formed from generic generalized constraints by combination, modification, and qualification. An example of a generalized constraint in GCL is ((Z isp R) and (Z, Y) is S) is unlikely.

By construction, the Generalized Constraint Language is maximally expressive. What this implies is that PNL is the largest subset of a natural language that admits precisiation. Informally, this implication serves as a basis for the conclusion that if a concept, X, cannot be defined in terms of PNL, then, in effect, it is undefinable or, synonymously, amorphic.

In this perspective, the highest level of definability hierarchy, which is the level above PNL-definability, is that of undefinability or amorphicity. A canonical example of amorphic concepts is that of causality. More specifically, it is not possible to construct a

general definition of causality such that given any two events A and B and the question, "Did A cause B?" the question could be answered based on the definition. Equivalently, given any definition of causality, it will always be possible to construct examples to which the definition would not apply or yield counterintuitive results.

In dealing with an amorphic concept, X, what is possible—and what we generally do—is to restrict the domain of applicability of X to instances for which X is definable. For example, in the case of the concept of a summary, which is an amorphic concept, we could restrict the length, type, and other attributes of what we want to summarize. In this sense, an amorphic concept may be partially definable or, p-definable, for short. The concept of p-definability applies to all levels of the definability hierarchy.

In essence, PNL may be viewed as a collection of ordered pairs of the form (p, $p^\times$), where p is a precisiable proposition in NL and $p^\times$ is a precisiation of p, that is, its translation in GCL. In this sense, PNL may be viewed as a dictionary in which p is an entry and $p^\times$ is its meaning.

In scientific theories, a concept, X, is almost always defined as a crisp (bivalent) concept, meaning that the denotation of X is a crisp set in its universe of discourse. In THD, a concept, X, is associated with a quintuple (X, U, QCS, DF(L), D(DF)) in which X is the concept; U is the space of objects to which X is applicable; QCS is the qualitative complexity scale associated with X; DF(L) is a definition of X in a language L; and D(DF) is the domain of DF, that is, the set of objects to which DF is applicable.

The concept of a qualitative complexity scale plays a key role in THD. Basically, the qualitative complexity scale, QCS, is a linear clustering, QCC_1, QCC_2, …, QCC_m, of qualitative complexity classes of objects in U such that: (a) objects in QCC_i are roughly equally complex in relation to the definition, DF, of X; and (b) objects in QCC_{i+1} have higher complexity than those in Q_i. For example, if X is the concept of volume, then QCC_2 may be class of objects like trees; and QCC_5 may be the class of objects like clothing. Each language in the definability hierarchy is associated with a critical threshold on the qualitative complexity scale such that the language cannot be applied to classes above the critical threshold.

As the lowest member of the definability hierarchy, the C language has a low expressive power, with the consequence that the associated critical threshold is near the low end of the of the qualitative complexity scale. In particular, the C language cannot be used to define fuzzy concepts. Thus, its use to define concepts which, in reality, are fuzzy concepts, leads to counterintuitive conclusions. An example is the conventional C-language-based definition of stability. Since stability, in general, is a matter of degree, its definition as a crisp concept leads to paradoxes similar to the ancient Greek sorites paradox. To define stability as a fuzzy concept—which in reality it is—what is needed is PNL. The same applies to the concept of causality. Thus, causality can be defined as a crisp concept only for complexity classes which lie close to the low end of the qualitative complexity scale.

Another important point is that almost every concept has some degree of amorphicity, with a concept such as causality being amorphic to a high degree. But even such basic concepts as volume, density, edge, derivative and optimality have domains of amorphicity which are apparent in many real-world settings. What this implies is that many basic concepts may require redefinition in terms of PNL.

Does PNL have a significant role to play in devising metrics of performance for intelligent systems? This is an issue that is not addressed in the brief sketch of the theory of hierarchical definability. But I have no doubt that it will, since the concept of intelligence is much too complex to lend itself to analysis through the use of existing bivalent-logic-based methods.